REMARKS

Status of Claims

Applicant respectfully requests reconsideration and allowance of all of the claims of the application. Claims 1-20 and 87-90 are presently pending. Claims 1-3, 7-12, 17, 19, 20 and 87-90 have been amended. No claims have been added, withdrawn or canceled. Claims 1, 11 and 87-90 are independent.

Statement of Substance of Interview

The Examiner graciously talked with me, the undersigned representative for the Applicant, on January 12, 2009. Applicant greatly appreciates the Examiner's willingness to talk. Such willingness is invaluable to both of us in our common goal of an expedited prosecution of this patent application.

During the interview, I discussed how the pending claims differed from the cited references, namely Brendel, Krause and Westberg. However, no agreement was reached regarding the patentability of the pending claims over the art of record.

Formal Request for an Interview

If the Examiner's reply to this communication is anything other than allowance of all pending claims, and if the Examiner feels that further discussion of the claims or the invention would advance prosecution of the application, then I formally request an interview with the Examiner. I encourage the Examiner to call me, the undersigned representative for the Applicant, so that we can discuss

this matter so as to resolve any outstanding issues quickly and efficiently over the phone.

Claim Amendments and Support

Without conceding the propriety of the rejections herein and in the interest of expediting prosecution, Applicant has amended claims 1-3, 7-12, 17, 19, 20 and 87-90 herein. Applicant amends these claims to clarify the claimed features. Such amendments are made to expedite prosecution and more quickly identify allowable subject matter, and should not be construed as further limiting the claimed invention in response to the cited references. The amendments to these claims are fully supported by the disclosure and do not constitute new matter. For example, support for the amendments to claims 1, 11 and 87-90 is found in the specification at least at paragraphs 0396 through 0417 and FIGS. 34-38 of the published present application, US2005/0055435.

FORMAL MATTERS

This section addresses any formal matters (e.g., objections) raised by the Examiner.

Claims

The Examiner objects to claim 3 for a typographical error. Applicant has amended claim 3, as shown above, to address the objection made by the Examiner, and to expedite prosecution. Entry of the amendment and withdrawal of the objection is respectfully requested.

Substantive Matters

Claim Rejections under § 103

The Office Action rejects claims 1-20 and 87-90 under § 103. For the reasons set forth below, the Office Action has not made a prima facie case showing that the rejected claims are obvious. Accordingly, Applicant respectfully requests that the § 103 rejections be withdrawn and the case be passed along to issuance.

The Office Action's rejections are based upon the following references alone or in combination:

- **Brendel:** *Brendel, et al.,* US Patent No. 5,774,660 (issued June 30, 1998);
- **Krause:** *Krause, et al.,* US Patent No. 6,047,323 (issued April 4, 2000); and
- **Westberg:** *Westberg, et al.,* US Patent No. 6,041,054 (issued April 4, 2000).

Overview of the Application

The Application describes a technology for a connection migrator that is configured to migrate connections away from the device. The connection migrator is capable of precipitating a compilation of protocol state for a connection across a protocol stack. The connection migrator is adapted to aggregate the compiled protocol state with data for the connection into an aggregated connection state and send the aggregated connection state toward a

target device. In an exemplary implementation, processor-executable instructions direct a device to perform actions including: obtaining at least a portion of a source/destination pair from a packet; accessing an encapsulation mapping table using the at least a portion of the source/destination pair to locate an encapsulation mapping entry; extracting a flow identifier from the encapsulation mapping entry; and replacing part of the packet with the flow identifier to produce an encapsulated packet.

Cited References

The Office Action cites Brendel as the primary reference in the obviousness-based rejections. The Office Action cites Krause and Westberg as secondary references in the obviousness-based rejections.

Brendel

Brendel describes a technology for load balancer that receives all requests from clients because they use a virtual address for the entire site. The load balancer makes a connection with the client and waits for the URL from the client. The URL specifies the requested resource. The load balancer waits to perform load balancing until after the location of the requested resource is known. The connection and URL request are passed from the load balancer to a second node having the requested resource. The load balancer re-plays the initial connection packet sequence to the second node, but modifies the address to that for the second node.

Krause

Krause describes a technology for operating a distributed STREAMS process on a multicomputer system composed of a cluster of nodes of one or more processors running an operating system having a file system and a STREAMS message-passing mechanism implementing network protocols, client-server applications, and STREAMS-based pipes. The distributed STREAMS process determines that it is operating within a cluster and transparently intercepts application open requests which are sent to a controlling thread (CT) created during node initialization. The CT determines whether the open is to occur on the local or a remote node and whether any cluster facility should be activated by examining major and minor numbers encoded within the file structure being opened. If a failure occurs, the CT and S-ICS detect this failure and transparently initiate error recovery by migrating failed components, if possible, to new node(s) within the cluster. This migration capability can also be used to provide load balancing within the cluster of distributed STREAMS.

Westberg

Westberg describes a technology for employing asynchronous transfer mode (ATM) adaption layer two (AAL2) minicells as a bearer. Bandwidth utilization and transmission efficiency may be further enhanced by mapping one or more data fields from the header portion of the IP data packets into one or more look-up tables and then transporting the look-up table addresses in the AAL2 minicell headers rather than the data associated with the one or more data fields in the IP data packet headers.

Obviousness Rejections

Lack of *Prima Facie* Case of Obviousness (MPEP § 2142)

Applicant disagrees with the Office Action's obviousness rejections. Arguments presented herein point to various aspects of the record to demonstrate that all of the criteria set forth for making a prima facie case have not been met.

Based upon Brendel

The Office Action rejects claims 1-6, 8, 11-16 and 88-89 under 35 U.S.C. § 103(a) as being unpatentable over Brendel in view of Krause. The Office Action rejects claims 7, 9, 10, 17-20, 87 and 90 as being unpatentable over Brendel in view of Krause, and further in view of Westberg. Applicant respectfully traverses the rejection of these claims and asks the Examiner to withdraw the rejection of these claims.

Independent Claims 1 and 88

Applicant submits that Brendel in combination with Krause and Westberg does not disclose, teach or suggest amended claim 1 because the combination of these references does not disclose the following elements as recited in this claim (with emphasis added):

> ...accepting a connection from a connecting device at a forwarder;

receiving data at the forwarder from the connecting device as a result of accepting the connection;

forwarding the data from the forwarder to a classifier;

determining, by the classifier, a second device for receiving the connection;

aggregating a connection state for the connection at the classifier by aggregating a protocol state of a first protocol stack and the data to constitute a binary blob;

sending the connection state from the classifier to the second device for injection into a second protocol stack at the second device by sending the binary blob including the protocol state and the data to the second device, whereby the connection is transferred to the second device;

in conjunction with sending the connection state, adding an entry to a mapping table maintained by the forwarder that indicates the second device as a destination for packets for the connection;

sending a mapping for a flow identifier to the second device based upon the entry in the mapping table;

receiving subsequent communications from the connecting device by the forwarder;

encapsulating the subsequent communications by the forwarder according to the entry in the mapping table of the forwarder by inserting the flow identifier into the encapsulated communications; and

receiving the encapsulated communications at the second device from the forwarder, wherein the flow identifier serves to identify a flow of encapsulated communications as being associated with the connection to the connecting device.

Brendel fails to teach or suggest, in conjunction with sending the connection state, adding an entry to a mapping table maintained by the

forwarder that indicates the second device as a destination for packets for the connection, or sending a mapping for a flow identifier to the second device based upon the entry in the mapping table. Instead, Brendel discusses that TCP state migration 120 is performed by the load balancer playing back packets received from the browser and stored by the load balancer (col. 12, lines 46-48). The load balancer then sends the browser's stored ACK packet to the assigned server, and the assigned server is then connected directly to the browser, having the same TCP state as was established with the load balancer (col. 12, lines 50-54). The load balancer then enters a pass-through state so that any further packets from the browser are passed through to the assigned server (col. 12, lines 59-62). Additional requests sent to the assigned server by the browser use a PUSH packet, PUSH(1), which is passed to the assigned server as PUSH(1)' (col. 12 line 64 through col. 13, lines 4 and FIGS. 11A-11B). Accordingly, Brendel fails to teach or suggest, in conjunction with sending the connection state, adding an entry to a mapping table maintained by the forwarder that indicates the second device as a destination for packets for the connection, or sending a mapping for a flow identifier to the second device based upon the entry in the mapping table;. Thus, Brendel also fails to teach or suggest encapsulating the subsequent communications by the forwarder according to the entry in the mapping table of the forwarder by inserting the flow identifier into the encapsulated communications, and receiving the encapsulated communications at the second device from the forwarder, wherein the flow identifier serves to identify a flow of encapsulated communications as being associated with the connection to the connecting device.

Westberg fails to make up for the shortcomings in Brendel discussed above. Westberg discusses that certain data fields of a header may be used for mapping a PPP (point-to-point) protocol ID and a session context/connection ID (col. 6, lines 22-26). More particularly, the PPP protocol ID may be mapped to the user-to-user information field, while the session context/connection ID may be mapped to the channel identification field (col. 6, lines 26-29). However, Westberg fails to teach or suggest, in conjunction with sending the connection state, adding an entry to a mapping table maintained by the forwarder that indicates the second device as a destination for packets for the connection, or sending a mapping for a flow identifier to the second device based upon the entry in the mapping table, as recited in Applicant's amended claim 1.

Krause fails to make up for the shortcomings in Brendel and Westberg discussed above. Consequently, Brendel in combination with Krause and Westberg does not teach or suggest Applicant's amended claim 1. Accordingly, Applicant respectfully asks the Examiner to withdraw the rejection of claim 1. Independent claim 88 includes limitations similar to those discussed above for claim 1, and is allowable under a similar rationale. Thus, claims 1 and 88 are in condition for allowance.

Dependent Claims 2-10

These claims ultimately depend upon independent claim 1. As discussed above, claim 1 is allowable. It is axiomatic that any dependent claim which

depends from an allowable base claim is also allowable. Additionally, some or all of these claims may also be allowable for additional independent reasons.

For example, amended dependent claim 3 includes sending the binary blob including the protocol state and the data asynchronously to a connection migrator component at the second device, wherein the connection migrator component is configured to receive the binary blob as a bundle, reassemble the connection state from the binary blob, and infuse the connection state into the second protocol stack at the second device. As discussed above, Brendel discloses that the load balancer replays the initial connection packet sequence to the assigned server. Thus, Brendel teaches only synchronous communication since the load balancer does not send the browser's stored ACK packet to the assigned server until the assigned server has returned a SYN/ACK packet in response to an initial SYN packet sent by the load balancer (col. 12, lines 46-54).

Krause discusses that the framework data is sent to the target controlling thread that will rebuild the stack framework (col. 68, lines 44-52). Then, the marshalling function is invoked for each stream component, and this structure is then sent to the target controlling thread (col. 68, lines 53-59). Thus, Krause does not send an aggregated connection state including the protocol state and the data as a binary blob asynchronously. Instead, Krause sends marshaled framework data in a separate transaction from sending of marshaled stream components (col. 68, lines 44-59), thereby carrying out multiple transactions. Thus, Krause combined with Brendel also fails to teach or suggest Applicant's claim 3.

Westberg fails to make up for these shortcomings in Brendel and Krause. Accordingly, Applicant respectfully submits that dependent claim 3 is in condition for allowance.

Furthermore, dependent claim 4 includes compiling the protocol state from the first protocol stack for use in offloading the connection state as a binary blob, wherein the compiled protocol state includes destination and source ports and IP addresses. As discussed above, Applicant respectfully submits that Brendel does not disclose compiling a protocol state that includes destination and source ports and IP addresses. Instead, Brendel discloses that packets are stored and played back to the assigned server (e.g., col. 12, lines 46-54).

Krause discusses that the framework data is sent to the target controlling thread that will rebuild the stack framework (col. 68, lines 44-52). Then, the marshalling function is invoked for each stream component, and this structure is then sent to the target controlling thread (col. 68, lines 53-59). Thus, Krause also does not teach or suggest compiling the protocol state from the first protocol stack for use in offloading the connection state as a binary blob, wherein the compiled protocol state includes destination and source ports and IP addresses. Westberg fails to make up for these shortcomings in Brendel and Krause. Applicant respectfully submits that dependent claim 4 is allowable over Brendel, Krause, Westberg and the other art of record, and accordingly, claim 4 is in condition for allowance.

Additionally, dependent claim 7 includes bundling the connection state with the mapping for the flow identifier that corresponds to the connection to produce

the binary blob, and transmitting the binary blob having the flow identifier mapping bundled therein from the classifier to the second device. As discussed above with respect to claim 1, Brendel, Krause and Westberg fail to teach or suggest transmitting mapping for a flow identifier. For example, Krause discusses that the framework data is sent to the target controlling thread that will rebuild the stack framework (col. 68, lines 44-52). Then, the marshalling function is invoked for each stream component, and this structure is then sent to the target controlling thread (col. 68, lines 53-59). Westberg discusses a flow identifier, but none of Brendel, Krause or Westberg teach or suggest bundling the connection state with the mapping for the flow identifier that corresponds to the connection to produce the binary blob, or transmitting the binary blob having the flow identifier mapping bundled therein from the classifier to the second device. Accordingly, Applicant respectfully submits that claim 7 is in condition for allowance.

Further, amended claim 10 includes forwarding subsequent communications for the connection to the second device using the flow identifier to encapsulate the subsequent communications, said encapsulated subsequent communications including the flow identifier in source and destination port fields of a TCP (Transmission Control Protocol) header. Brendel and Krause fail to teach or suggest this aspect of Applicant's invention. Westberg teaches using an IP/PPP data packet header to include a session context/connection ID (col. 6, lines 4-62. However, Westberg fails to teach or suggest that a flow identifier is included in source and destination port fields of a TCP header. Accordingly,

Applicant respectfully submits that dependent claim 10 is in condition for allowance.

The remaining dependent claims not discussed above can also be distinguished from the combination of Brendel, Krause and Westberg, and Applicant respectfully submits that these claims are also in condition for allowance.

Independent Claims 11 and 89

Applicant submits that Brendel in combination with Krause and Westberg does not disclose, teach or suggest amended claim 11 because the combination of these references does not disclose the following elements as recited in this claim (with emphasis added):

> ...accepting a connection from a connecting device by a forwarder at the first device;
>
> receiving data at the first device as a result of accepting the connection;
>
> aggregating, by a classifier at the first device, a connection state for the connection at the first device by aggregating a protocol state of a first protocol stack and the received data to constitute an aggregated connection state;
>
> **sending the aggregated connection state including the protocol state and the received data asynchronously from the first device to the second device;**

receiving the aggregated connection state asynchronously at the second device, whereby the aggregated connection state comprised of the protocol state and the received data is received intact at the second device;

injecting the aggregated connection state for the connection into a network stack at the second device;

in conjunction with sending the aggregated connection state, sending a mapping for a flow identifier from the first device to the second device, the flow identifier for identifying encapsulated packets received from the forwarder;

continuing the connection at the second device using the injected connection state;

receiving subsequent communications from the connecting device by the forwarder;

encapsulating the subsequent communications by the forwarder by inserting the flow identifier into the encapsulated communications according to a mapping table maintained by the forwarder; and

receiving the encapsulated communications at the second device from the forwarder, wherein the flow identifier serves to identify a flow of encapsulated communications as being associated with the connection to the connecting device according to the mapping for the flow identifier received from the first device.

Applicant's amended claim 11 includes, in conjunction with sending the aggregated connection state, sending a mapping for a flow identifier from the first device to the second device, the flow identifier for identifying encapsulated packets received from the forwarder. As discussed above with respect to claim 1, the combination of Brendel, Krause and Westberg fails to teach or suggest this aspect of amended claim 11. Accordingly claim 11 is allowable for this reason.

Furthermore, as recited in amended claim 11, the connection state is aggregated by aggregating a protocol state of a first protocol stack and the data, and the aggregated connection state is sent and received asynchronously. As discussed above, Brendel discloses that the load balancer replays the initial connection packet sequence to the assigned server. Thus, Brendel teaches only synchronous communication since the load balancer does not send the browser's stored ACK packet to the assigned server until the assigned server has returned a SYN/ACK packet in response to an initial SYN packet sent by the load balancer (col. 12, lines 46-54).

Krause discusses that the framework data is sent to the target controlling thread that will rebuild the stack framework (col. 68, lines 44-52). Then, the marshalling function is invoked for each stream component, and this structure is then sent to the target controlling thread (col. 68, lines 53-59). Thus, Krause does not send an aggregated connection state including the protocol state and the received data asynchronously. Instead, Krause sends marshaled framework data in a separate transaction from sending of marshaled stream components (col. 68, lines 44-59), thereby carrying out plural transactions for sending framework data and stream components. Under Applicant's claim 11, however, and an aggregated connection state including the protocol state and the received data is sent. Applicant respectfully submits that Krause does not teach or suggest this.

Westberg fails to make up for the shortcomings in Brendel and Krause discussed above. Accordingly, Applicant asks the Examiner to withdraw the rejection of claim 11. Independent claim 89 includes limitations similar to those

discussed above for claim 11, and is allowable under a similar rationale. Accordingly, Applicant respectfully submits that claims 11 and 89 are in condition for allowance.

Dependent Claims 12-20

These claims ultimately depend upon independent claim 11. As discussed above, claim 11 is allowable. It is axiomatic that any dependent claim which depends from an allowable base claim is also allowable. Additionally, some or all of these claims are also allowable for additional independent reasons as discussed above relative to dependent claims 2-10.

New Independent Claims 87 and 90

Independent claims 87 and 90 include limitations similar to those discussed above with respect to claims 1, 11, 88 and 89, and are allowable under a similar rationale. Furthermore, claims 87 and 89 also include the following limitations (with emphasis added):

> ...receiving a connection request by a forwarder at the first device from a client device;
>
> accepting the connection request at the first device by sending an acknowledgment packet to the client device in response to the connection request;
>
> receiving data for the connection at the first device from the client device;

determining, by a classifier at the first device, the second device to which to migrate the connection from among a plurality of second devices, based upon the received data;

compiling a protocol state for the connection from a first protocol stack at the first device;

aggregating a connection state for the connection by aggregating the compiled protocol state and the received data to constitute a binary blob;

bundling a mapping for a flow identifier into the binary blob, wherein the flow identifier is used by the second device in identifying a source of encapsulated packets sent to the second device as subsequent communications from the first device corresponding to the connection;

sending the connection state from the first device by asynchronously sending the binary blob to the second device;

receiving the connection state as the bundled binary blob at the second device;

unbundling the aggregated connection state and the mapping for the flow identifier at a level that is below a second protocol stack at the second device;

injecting the connection state by the connection migrator component into the second protocol stack at the second device;

in conjunction with sending the connection state, adding an entry to a mapping table maintained by the forwarder that indicates the second device as a destination for subsequent communications for the connection, wherein the entry corresponds to the mapping for the flow identifier sent to the second device;

continuing the connection at the second device using the injected connection state;

receiving packets as subsequent communications from the client device by the forwarder at the first device;

encapsulating the packets by the forwarder by inserting the flow identifier into the encapsulated packets according to the entry in the mapping table maintained by the forwarder, wherein the flow identifier is encoded in source and destination fields of a TCP (Transmission Control Protocol) header of the encapsulated packet; and

receiving the encapsulated packets at the second device from the forwarder, wherein the flow identifier serves to identify a flow of encapsulated packets received by the second device from the forwarder as being associated with the connection with the client device.

Thus, Applicant's invention includes bundling a mapping for a flow identifier into the binary blob, wherein the flow identifier is used by the second device in identifying a source of encapsulated packets sent to the second device as subsequent communications from the first device corresponding to the connection, sending the connection state from the first device by asynchronously sending the binary blob to the second device, receiving the connection state as the bundled binary blob at the second device, and unbundling the aggregated connection state and the mapping for the flow identifier at a level that is below a second protocol stack at the second device before injection into the second protocol stack.

Brendel, Krause and Westberg fail to teach or suggest transmitting mapping for a flow identifier bundled into a binary blob with an aggregated connection state. For example, Krause discusses that the framework data is sent to the target controlling thread that will rebuild the stack framework (col. 68, lines 44-52). Then, the marshalling function is invoked for each stream

component, and this structure is then sent to the target controlling thread (col. 68, lines 53-59). Westberg discusses a flow identifier, but none of Brendel, Krause or Westberg teach or suggest bundling a mapping for a flow identifier into the binary blob, wherein the flow identifier is used by the second device in identifying a source of encapsulated packets sent to the second device as subsequent communications from the first device corresponding to the connection, as recited in Applicant's claims 87 and 90. Accordingly, Applicant respectfully submits that claims 87 and 90 are in condition for allowance.

Conclusion

All pending claims are in condition for allowance. Applicant respectfully requests reconsideration and prompt issuance of the application. If any issues remain that prevent issuance of this application, the **Examiner is urged to contact me before issuing a subsequent Action**. Please call or email me at your convenience.

Respectfully Submitted,

Lee & Hayes, PLLC
Representatives for Applicant

/Colin D. Barnitz/ Dated: January 22, 2009

Colin D. Barnitz (colin@leehayes.com)
Registration No. 35061
Reviewer/Supervisor Emmanuel Rivera (emmanuel@leehayes.com)
Registration No. 45,760

Customer No. **22801**

Telephone: (512) 505-8167
Facsimile: (509) 323-8979
www.leehayes.com